# An Analysis of Average Cell Variation within Voronoi Diagrams

James deLeon – 1001592708

July 31, 2025

## 1 Introduction

Voronoi Diagrams are versatile geometric structures with a range of applications in geographical path finding [1], computer graphics [2], image approximation [3] and even solid-state physics [4]. The inspiration for these tessellated structures comes from naturally occurring environments, as seen in Figure 1. The simplicity of the polygonal formations can easily be captured as a mathematical object, as defined below in Def. 1.1.

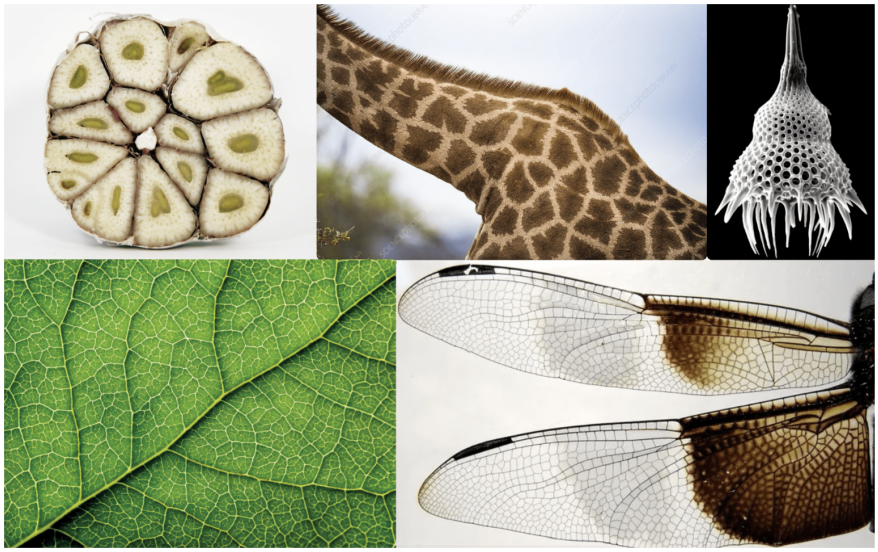

Figure 1: Voronoi tessellations found in nature [5].

**Definition 1.1** ([6])**.** Let $S = \{s_1, s_2, s_3, \cdots, s_n\} \subset \mathbb{R}^2$ be a finite set of sites in a Euclidean space. Then, the Voronoi cell $V_i$ associated with site $s_i$ is defined as follows for some point in space $x \in \mathbb{R}^2$:

$$V_i = \left\{x \in \mathbb{R}^2 : \|x - s_i\| \leq \|x - s_j\| \text{ for all } j \neq i\right\}$$

where $\|\cdot\|$ denotes the norm operator in Euclidean space between point $x$ and $s_i$. As an interpretation, every coordinate in Euclidean space is assigned to the site $s_i$ that minimizes the Euclidean distance between the point $x$ and the site $s_i$; this collection of points in Euclidean space assigned to a site $s_i$ is denoted as a Voronoi cell.

Now, with the structure of a Voronoi diagram defined, we want to examine the relationship between the distribution of cell areas present in a $N \times N$ pixel diagram composed of $p$ Voronoi sites. We define two populations of Voronoi diagrams such that their respective distributions of Voronoi sites is uniformly and Normally random. Figure 2 below shows a series of diagrams with an increasing count of Voronoi sites with a uniform random distribution.

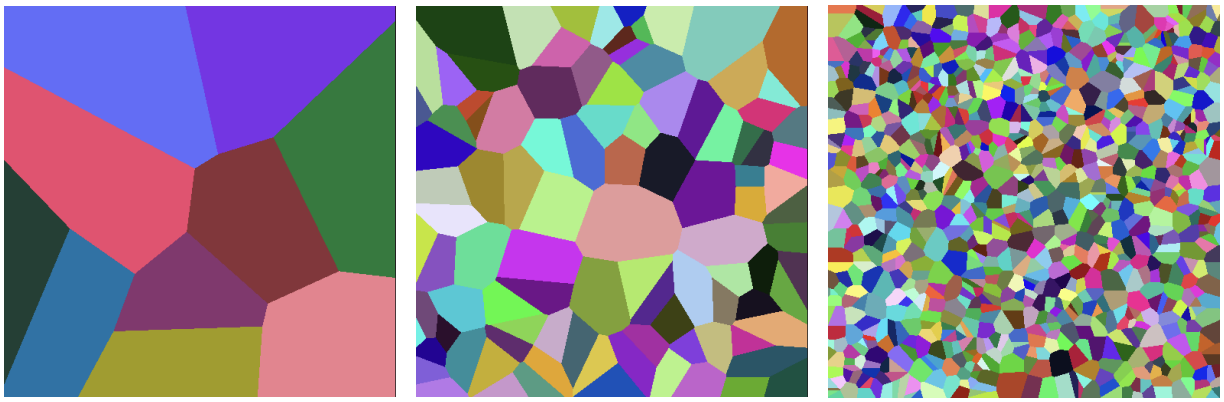

Figure 2: Voronoi diagram ($512 \times 512$ px) with a uniformly random distribution of a) 10, b) 100, and c) 1000 sites.

To distinguish each cell in the diagram, each site is given a random RGB color, such that all neighboring points on the canvas are also assigned the color, following Def. 1.1. Likewise, Figure 3 below shows a series of diagrams with an increasing count of Voronoi sites with a Normal random distribution.

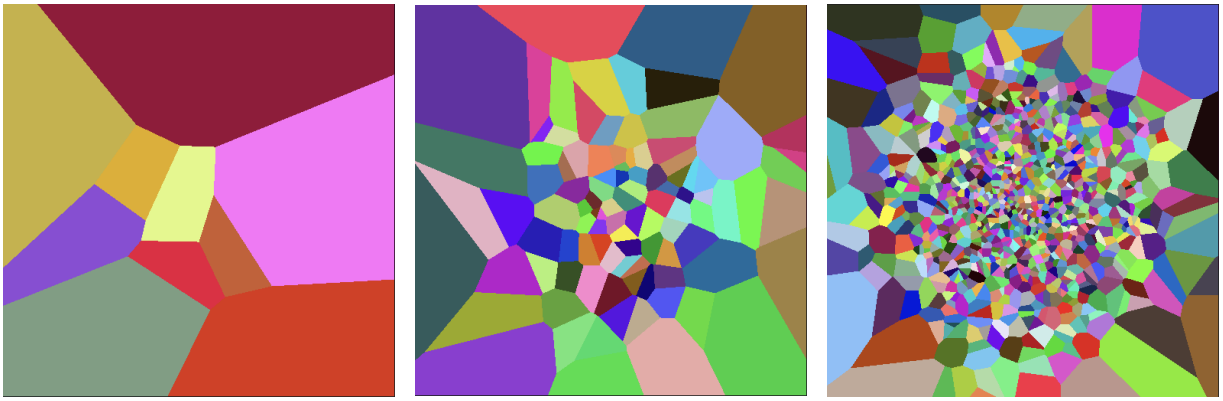

Figure 3: Voronoi diagram ($512 \times 512$ px) with a normal random distribution of a) 10, b) 100, and c) 1000 sites.

Notice that as the number of sites in a diagram increases, the overall distributions that govern the placement of the sites are more evidently different. For instance, a diagram with Normally distributed sites have a denser cluster of sites around the center of the canvas with sparser, and therefore larger, cell placements near the edges of the canvas. Now, it can observed (and maybe later proven) that a partition of cells on a finite canvas creates a deterministic value for the mean cell area of a single diagram. Specifically, let $\bar{A}$ denote the mean cell area for a population of $N \times N$ pixel diagram composed of $p$ sites. Then,

$$\bar{A} = \frac{N^2}{p}$$

There are refined methods that can computationally generate Voronoi diagrams, as will be further discussed in Section 2, but the component of analysis here will be how the diagram structures are altered specifically between uniformly distributed and normally distributed sites on a 2D canvas. Even though a population of Voronoi diagrams have a deterministic mean cell area, the variation of individual cell areas can be measured as a random variable. We will compare the variation of cell areas between uniformly distributed sites and Normally distributed sites. Let $\mu_1$ denote the population mean variance of $512 \times 512$ pixel Voronoi diagrams with $p = 256$ Normally distributed sites, and let $\mu_2$ denote the population mean variance of diagrams with $p = 256$ uniformly distributed sites. Then, this report proposes an alternative hypothesis that there is significant evidence, with $\alpha = 0.05$, there a difference in mean variance among Voronoi cell areas

between the two populations:

$$H_0 : \mu_1 - \mu_2 = 0$$

$$H_a : \mu_1 - \mu_2 \neq 0$$

In order to reject the above $H_0$, we need to utilize a two-sample t-test to find a P-value such that $2P(t > |t_0|) < \alpha$, as discussed further in Section 2.

## 2 Methodology

Many algorithms have been developed to create a contour of Voronoi cells given a set of sites placed on a 2D canvas, the most notable of which is Fortune's sweepline algorithm [7].To efficiently measure the cell area of a diagram however, requires that our diagram contain information about the entire metric space of the diagram and not just the line segments outlining the Voronoi cells. So, we look at a divide-and-conquer algorithm [8] with similar computational complexity as Fortune's algorithm has been developed to emphasize the size of the occupied Voronoi cells, thus allowing for the tabulation of combined areas assigned to a specific Voronoi site. This divide-and-conquer method uses the Quadtree [9] data structure, which recursively assigns a pixel to it's corresponding Voronoi site that minimizes the Euclidean distance between the two points. Figure 4 shows how the diagrams are structured with outlines of Quadtree sectors.

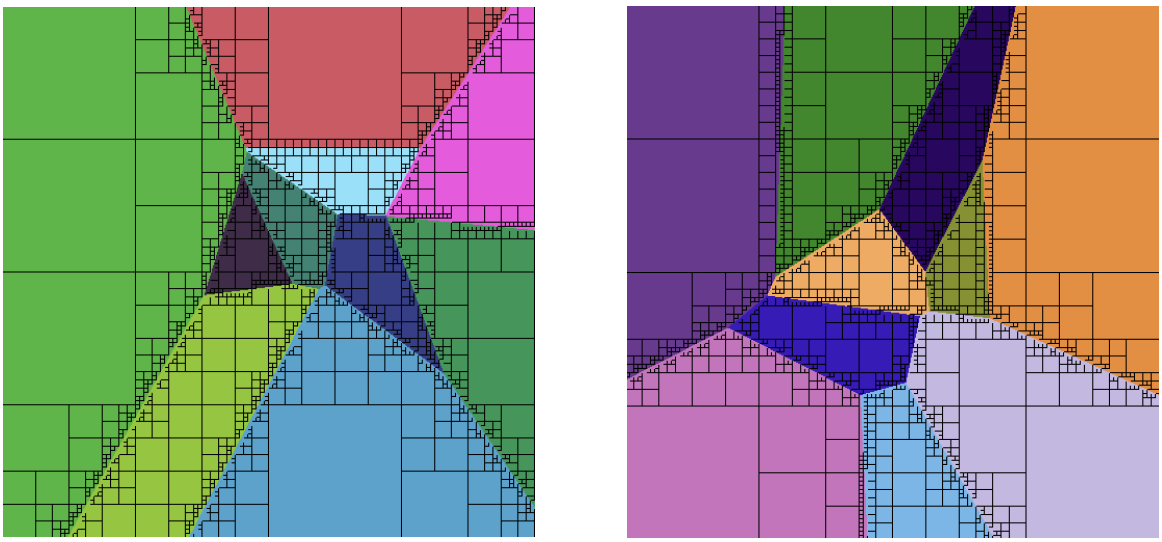

Figure 4: A demonstration of a Voronoi diagrams ($512 \times 512$ px, 10 sites) composed of a quadtree structure.

This divide-and-conquer method analyzes the Euclidean distance from each of the four pixels in a single Quadtree sector; if four corner all pixels share the same color, then the recursion ends for the sector, otherwise the recursion will continue until a Quadtree leaf node consists of four sectors each with area 1 sq. px. This method provides a noticeable increased computational efficiency with a few Voronoi sites, since Quadtree sectors can cover larger monochromatic areas on the diagram.

For this analysis, we will be examining the average deviation for both uniformly and Normally distributed sites, which can be included in the generation of the diagram as a procedural random number generator, as mentioned in Appendix 6.1. Comparing two populations of Voronoi diagrams, it would be suiting to proceed with a two-sample t-test. We now define our sample statistics. Let $n$ denote the sample size, then our sample mean, $\bar{x}$, is defined as –

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{1}$$

Likewise, our sample variance, $s^2$, take the following form,

$$s^2 = \frac{1}{n-1}\sum_{i=1}^{n} (x_i - \bar{x})^2 \tag{2}$$

Note that since we're examining the population variance between two random distributions of Voronoi sites, our sample mean and sample variance are with respect to their corresponding population variations. In order to reject our null hypothesis, we must calculate a P-value such that $2P(t > |t_0|) < \alpha$, where $t_0$ is our two-sample t-statistic defined as follows,

$$t_0 = \frac{\bar{x}_1 - \bar{x}_2 - (\mu_1 - \mu_2)}{\sqrt{\dfrac{s_1^2}{n_1} + \dfrac{s_2^2}{n_2}}} \tag{3}$$

and corresponding confidence interval defined as follows,

$$(\bar{x}_1 - \bar{x}_2) \pm t^* \sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}} \tag{4}$$

In this experiment, it is assumed that the two populations do not have equal standard deviations. That is, the standard deviation in the distribution of average variance of cell areas are not equal

between diagrams with uniformly and Normally distributed Voronoi sites. As mentioned before, the divide-and-conquer method allows for large diagrams to be generated within several milliseconds, so this experiment will produce a sampling size of $n_1 = n_2 = 1001$ for both cases. Given our assumptions of the population deviation being an unknown parameter and unequal between populations, we can use a t-distribution [10] to examine the critical value at 1000 degrees of freedom. Specifically,

$$\text{df} = \min(n_1 - 1, n_2 - 1) = 1001 - 1 = 1000$$

Now, for 95% confidence, Table D [10] lists a critical value of $t^* = 1.962$ for 1000 degrees of freedom.

## 3 Results

Below shows the results from the simulation.

| | $\bar{x}$ (sq. px) | $s$ (sq. px) | $n$ |
|---|---|---|---|
| Normal Distribution ($\mu_1$) | 322081 | 44705 | 1001 |
| Uniform Distribution ($\mu_2$) | 319261 | 44219 | 1001 |

Table 1: Voronoi cell area statistics

Calculating our 95% confidence interval from Eq. 4, we have –

$$(322081 - 319261) \pm 1.962\sqrt{\frac{(44705)^2}{1001} + \frac{(44219)^2}{1001}} = (2820 \pm 3899) \text{ sq. px}$$

This interval indicates that if we repeat this sampling process many times, then the true difference in mean variance will be contained between -1079 sq. pixels and 6719 sq. pixels 95% of the time. The lower bound extending into the negative region indicates that the Uniform distribution of Voronoi sites presents a greater average variation of Voronoi cell areas, while the upper bound indicates a greater average variation of cell areas from Normally distributed sites. This is not a good sign for our significance test measuring a difference between the variations in the two populations since a mean difference of 0 also lies in the interval. This presents evidence of a similarity between the two distributions, as seen in Figure 5 below.

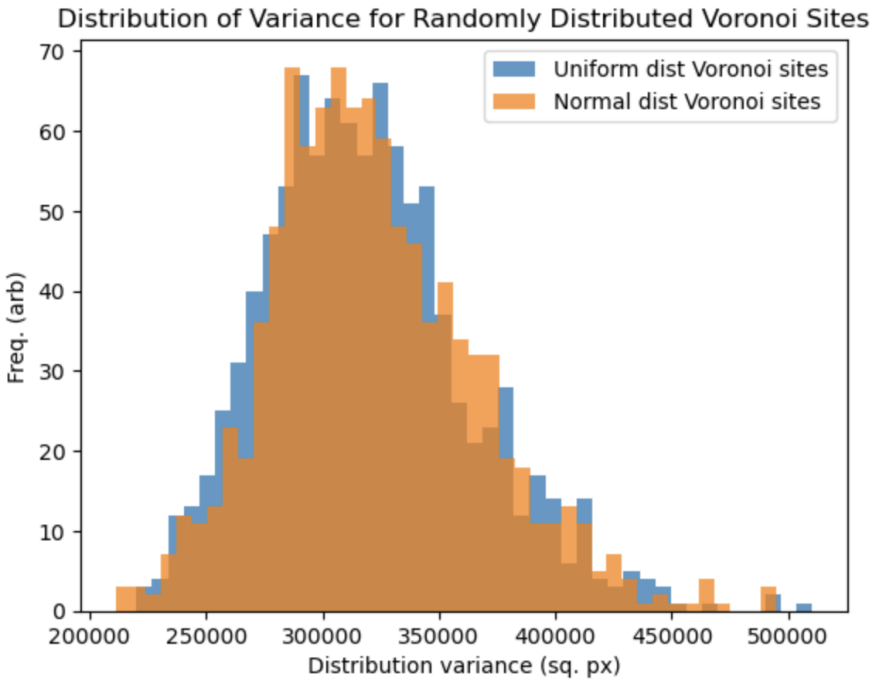


Figure 5: Sampling distribution of Voronoi cell area variance

With a significance level of $\alpha = 0.05$, we can calculate our test statistic using Eq. 3, as follows –

$$t_0 = \frac{322081 - 319261}{\sqrt{\frac{(44705)^2}{1001} + \frac{(44219)^2}{1001}}} \approx 1.42$$

Using Table D [10], we can see that since our critical value is 1.962, our test statistic is less than the critical value. Specifically,

$$2P(t > 1.282) < 2P(t > 1.42) < 2P(t > 1.646)$$

$$0.10 < P(t > 1.42) < 0.20$$

Therefore, we fail to reject the null hypothesis and conclude that there is not enough evidence presented here to assume there is a significant difference between random site distributions.

# 4 Conclusion

Figure 6 below shows another angle to the sampling distribution. We can see that although the sampling distribution is approximately Normal, there exist some outliers on the right tail of the distribution, indicating that the distribution has an influence of skewing to the right. This makes sense because as the number of sites on a diagram decrease, the outer fringes of the diagram have a greater variation of cell areas than the central regions. This phenomenon can connect to the Central Limit Theorem; as the number of Voronoi sites increases, the average variation of cell areas between distributions is not significantly different.

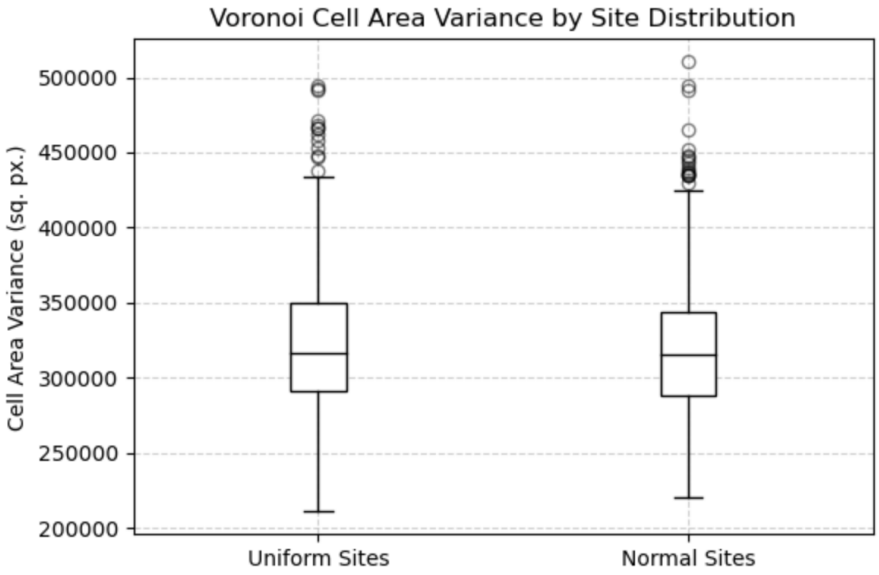


Figure 6: Distribution profiles of Voronoi cell area variance

Overall, more studies have to be conducted on the distributions of Voronoi diagrams in order to measure a statistically significant difference between site distributions. From a practical standpoint, if a Voronoi diagram is to be generated with a large number of Voronoi sites, then it can be assumed that the method by which the sites are generated (whether uniformly or Normally) does not produce a statistically significant difference between the variation in cell areas between diagrams.

## 5 Acknowledgements

Voronoi diagrams are very fascinating structures, and I believe there are many more discoveries to be made incorporating these structures. I have been developing the divide-and-conquer algorithm used in [8] independently in attempt to perform more experiments on Voronoi diagrams applied in image approximation and computer graphics. More information regarding this project's source code can be found here: `https://github.com/jdeleon-py/Project_ComputationalGeometries/tree/main/voronoi_diagrams`

## References

[1] Mark De Berg. *Computational geometry: algorithms and applications*. Springer Science & Business Media, third edition, 2000.

[2] Marcelo Walter, Alain Fournier, and Daniel Menevaux. Integrating shape and pattern in mammalian models. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 317–326, 2001.

[3] Koichi Kise, Akinori Sato, and Motoi Iwata. Segmentation of page images using the area voronoi diagram. *Computer Vision and Image Understanding*, 70(3):370–382, 1998.

[4] Emanuel A Lazar, Jiayin Lu, and Chris H Rycroft. Voronoi cell analysis: The shapes of particle systems. *American Journal of Physics*, 90(6):469–480, 2022.

[5] Nisha Arya. A quick overview of voronoi diagrams, 2022.

[6] Adam Dobrin. A review of properties and variations of voronoi diagrams. *Whitman College*, 10(1.453):9156, 2005.

[7] Steven Fortune. A sweepline algorithm for voronoi diagrams. In *Proceedings of the second annual symposium on Computational geometry*, pages 313–322, 1986.

[8] Elijah Smith, Christian Trefftz, and Byron DeVries. A divide-and-conquer algorithm for computing voronoi diagrams. In *2020 IEEE International Conference on Electro Information Technology (EIT)*, pages 495–499. IEEE, 2020.

[9] Raphael A Finkel and Jon Louis Bentley. Quad trees a data structure for retrieval on composite keys. *Acta informatica*, 4(1):1–9, 1974.

[10] David S Moore, George P McCabe, and Bruce A Craig. *Introduction to the Practice of Statistics*. W.H Freeman, 9th edition, 2016.

[11] George EP Box and Mervin E Muller. A note on the generation of random normal deviates. *The annals of mathematical statistics*, 29(2):610–611, 1958.

# 6 Appendix

## 6.1 Generation of Random Distributions

To map two uniformly random coordinates to normally random coordinates, we use the Box and Muller method [11]. Specifically, we let $U_x$, $U_y \in \mathbb{Z}$ be independent uniformly random variables representing corresponding 2D coordinates on a diagram such that the origin is the center of the diagram. Then, a mapping to a Normal distribution $N_x, N_y \in \mathbb{Z}$ with zero mean and unit variance will take the following form:

$$N_x = \sqrt{-2 \ln U_x} \cdot \cos(2\pi U_y) \tag{5}$$

$$N_y = \sqrt{-2 \ln U_x} \cdot \sin(2\pi U_y) \tag{6}$$

For the purpose of this investigation, the mappings $N_x, N_y$ are to be rescaled such that the two-dimensional Gaussian profile can extend over the entire $512 \times 512$ canvas. This implementation of the generation of Normal numbers ensures that 99.7% of $N_x, N_y$ values are contained within the canvas dimensions. Note that the few generated coordinates that fall outside of the canvas domain will be clamped to either the min or max value of the canvas's dimensions, accordingly.